5

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/994,599	11/19/2001	Kenneth Y. Ogami	CYPR-CD01198M	5425

7590 04/02/2007

WAGNER, MURABITO & HAO LLP
Third Floor
Two North Market Street
San Jose, CA 95113

EXAMINER	
BONSHOCK, DENNIS G	
ART UNIT	PAPER NUMBER
2173	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	04/02/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No. 09/994,599	Applicant(s) OGAMI ET AL.
	Examiner Dennis G. Bonshock	Art Unit 2173

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 24 January 2007.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-27 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-27 is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

Final Rejection

Response to Amendment

1. It is hereby acknowledged that the following papers have been received and placed on record in the file: Amendment as received on 1-24-2007.

2. Claims 1-27 have been examined.

Status of Claims:

3. Claims 1-5, 7, 9-14, 16, 18-23, 25, 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nilsson, *Introduction to PSpice Manual for Electric Circuits Using OrCad Release 9.1*, Lawman et al., Patent #5,673,198, hereinafter Lawman, and Williams, Patent #6,631,508.

4. Claims 6, 15, and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nilsson, Lawman, Williams, and Hsu, Patent #6,138,270.

5. Claims 8, 17, and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nilsson, Lawman, Williams, and McDonald et al., Patent #6,530,065, hereinafter McDonald.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-5, 7, 9-14, 16, 18-23, 25, 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nilsson, *Introduction to PSpice Manual for Electric Circuits Using OrCad Release 9.1*, Lawman et al., Patent #5,673,198, hereinafter Lawman, and Williams, Patent #6,631,508.

3. With regard to claim 1, which teaches a method of API generation for an electronic circuit comprising: displaying a graphical user interface through which a user can initiate generation of the API, Nilsson teaches on page 1, a program which provides an interface for the use in simulating electronic circuits. With regard to claim 1, further teaching selecting a component from a plurality of components for placement in the electronic circuit, the component for executing an implementable embedded function in the electronic circuit, Nilsson teaches, on pages 4 and 7, the process of selecting a component (place/part) and placing it in the schematic, the part having embedded functionality which is executed upon simulation/analysis. With regard to claim 1, further teaching configuring the selected component via the GUI, Nilsson teaches, on pages 4 and 5, the user configuring the selected part. With regard to claim 1, further teaching storing descriptive data relative to said selected component and said configuration, Nilsson teaches, on pages 4 and 5, the user configuring the selected part, adding descriptive data and clicking ok to store this information in association with the part. With regard to claim 1, teaching utilizing the interface to access the descriptive data, Nilsson teaches, on pages 4 and 5, double clicking on a part to see descriptive data. With regard to claim 1, further teaching initiating the GUI to invoke a processing of the descriptive data causing an automatic generation of the new API, in response to a user

input, the API for controlling the function of the component in the electronic circuit, Nilsson teaches, on pages 9-12, the system processing the descriptive data of the completed circuit and automatically generating an Interface (such as the net list, later displayed in Pspice A/D window (see page 7 and 11)), in response to the user input selecting the Pspice/Run menu option.

Nilsson teaches a system in which a graphical user interface is started with an application program, and upon a schematic circuit being drawn, via user input in the graphical user interface, a netlist is generated by analyzing the schematic circuit elements and their underling attributes, this netlist is then analyzed and provided for display on the GUI (see pages 7 and 11), however, he doesn't specifically teach an API comprising a device interface used to customize functionality of the electronic circuit; and interrupt activity framework for source programming and controlling the embedded function of the component in the electronic circuit, through user interaction with said new API. Lawman teaches a system for providing real time design of an electronic circuit via user input through a graphical user interface creating or modifying a logic circuit (see abstract and column 5, lines 30-42), similar to that of Nilsson, but further teaches that when a user adds a circuit element to the electronic circuit it causes corresponding elements to be added to the netlist, these take the form of API messages (forming program code based on functions assigned to components in the circuit diagram). After implementation, API messages are set back to the schematic entry system for display as feedback on the GUI (see column 5, lines 36-42 and column 6, lines 14-55). It would have been obvious to one of ordinary skill in the art, having the

teachings of Nilsson and Lawman before him at the time the invention was made to modify electronic circuit design program of Nilsson to include the API functionality of Lawman. One would have been motivated to make such a combination because the systems implement a similar means of generating an interface via graphical programming, Lawman further specifies user input causing an API to generate the GUI, via initiating processing, which would be an effective way of GUI generation for Nilsson.

Though it seems that the invention of Nilsson and Lawman modifies an API forming a new API (see column 6, lines 25-34 of Lawman), neither specifically state the forming of the new API. Williams teaches a system for designing a circuit through functional parts (see column 2, lines 56-65), similar to that of Nilsson and Lawman, but further teaches the creation of an API that defines the functional characteristics of the circuit and teaches allowing user manipulation of objects defined in the API (see column 2, line 56 through column 3, line 16). It would have been obvious to one of ordinary skill in the art, having the teachings of Nilsson, Lawman, and Williams before him at the time the invention was made to modify the circuit design systems of Nilsson and Lawman to include the actual creation of a new API, did Williams. One would have been motivated to make such a combination because this allows for greater portability of the designed circuit, in all references a user is creating a circuit design and allowing a user to interface with the created circuit, effectively creating an API (interface to an program representing the electronic circuit).

4. With regard to claims 2, 11, and 20, which teach the configuring the selected components comprises placing the selected component in the electronic circuit, Nilsson

teaches, on page 4, configuring the circuit by selecting a component (place/part) and placing it in the schematic.

5. With regard to claims 3, 12, and 21, which teach the configuring the selected component further comprising setting parameters of the selected component, the parameters relative to the function of the component and to the electronic circuit, Nilsson teaches, on pages 4 and 5, the user configuring the selected part by setting parameters that are relative to a function of the component in the circuit.

6. With regard to claims 4, 13, and 22, which teach configuring the selected component comprising selecting pin values for connecting the component to the electronic circuit, Nilsson teaches, on page 46, connecting the op amp to VCC+ and VCC-, which in turn correspond to power supplies displayed connected over ground.

7. With regard to claims 5, 14, and 23, which teach the API comprising header files, Nilsson teaches, on page 12, the inclusion of a definition file.

8. With regard to claims 7, 16, and 25, which teach the API comprising include files, Nilsson teaches, on page 12, the inclusion of an include file.

9. With regard to claims 9, 18, and 27, which teach the interface comprising a plurality of windows, Nilsson teaches, on pages 5, 9, and 12, the interface comprising a plurality of windows.

10. With regard to claim 10, which teaches a computer readable medium having computer useable program code causing the computer to perform: displaying a graphical user interface through which a user can initiate generation of the API, Nilsson teaches on page 1, a program which provides and API for the use in simulating

electronic circuits. With regard to claim 10, further teaching selecting a component from a plurality of components for placement in the electronic circuit, the component for executing an implementable embedded function in the electronic circuit, Nilsson teaches, on pages 4 and 7, the process of selecting a component (place/part) and placing it in the schematic, the part having embedded functionality which is executed upon simulation/analysis. With regard to claim 10, further teaching configuring the selected component via the GUI, Nilsson teaches, on pages 4 and 5, the user configuring the selected part. With regard to claim 10, further teaching storing descriptive data relative to said selected component and said configuration, Nilsson teaches, on pages 4 and 5, the user configuring the selected part, adding descriptive data and clicking ok to store this information in association with the part. With regard to claim 10, teaching utilizing the interface to access the descriptive data, Nilsson teaches, on pages 4 and 5, double clicking on a part to see descriptive data. With regard to claim 10, further teaching initiating the GUI to invoke a processing of the descriptive data causing an automatic generation of the API, in response to a user input, the API for controlling the function of the component in the electronic circuit, Nilsson teaches, on pages 9-12, the system processing the descriptive data of the completed circuit and automatically generating an API (such as the net list, later displayed in Pspice A/D window (see page 7 and 11)), in response to the user input selecting the Pspice/Run menu option.

Nilsson teaches a system in which a graphical user interface is started with an application program, and upon a schematic circuit being drawn, via user input in the

graphical user interface, a netlist is generated by analyzing the schematic circuit elements and their underling attributes, this netlist is then analyzed and provided for display on the GUI (see pages 7 and 11), however, he doesn't specifically teach an API comprising a device interface used to customize functionality of the electronic circuit; and interrupt activity framework for source programming and controlling the embedded function of the component in the electronic circuit, through user interaction with said new API. Lawman teaches a system for providing real time design of an electronic circuit via user input through a graphical user interface creating or modifying a logic circuit (see abstract and column 5, lines 30-42), similar to that of Nilsson, but further teaches that when a user adds a circuit element to the electronic circuit it causes corresponding elements to be added to the netlist, these take the form of API messages (forming program code based on functions assigned to components in the circuit diagram). After implementation, API messages are set back to the schematic entry system for display as feedback on the GUI (see column 5, lines 36-42 and column 6, lines 14-55). It would have been obvious to one of ordinary skill in the art, having the teachings of Nilsson and Lawman before him at the time the invention was made to modify electronic circuit design program of Nilsson to include the API functionality of Lawman. One would have been motivated to make such a combination because the systems implement a similar means of generating an interface via graphical programming, Lawman further specifies user input causing an API to generate the GUI, via initiating processing, which would be an effective way of GUI generation for Nilsson.

Though it seems that the invention of Nilsson and Lawman modifies an API forming a new API (see column 6, lines 25-34 of Lawman), neither specifically state the forming of the new API. Williams teaches a system for designing a circuit through functional parts (see column 2, lines 56-65), similar to that of Nilsson and Lawman, but further teaches the creation of an API that defines the functional characteristics of the circuit and teaches allowing user manipulation of objects defined in the API (see column 2, line 56 through column 3, line 16). It would have been obvious to one of ordinary skill in the art, having the teachings of Nilsson, Lawman, and Williams before him at the time the invention was made to modify the circuit design systems of Nilsson and Lawman to include the actual creation of a new API, did Williams. One would have been motivated to make such a combination because this allows for greater portability of the designed circuit, in all references a user is creating a circuit design and allowing a user to interface with the created circuit, effectively creating an API (interface to an program representing the electronic circuit).

11. With regard to claim 19, which teaches a computer system comprising: a bus, a display device coupled to the bus, a memory unit coupled to the bus, and a processor coupled to the bus, Nilsson teaches on page vii, the system being implemented on a Computer Aided Design, where computers are known in the art to comprise a memory unit and a processor and an associated bus connecting the components, the invention further discloses the display circuit related data on a display screen. With regard to claim 19, further teaching a method of API generation for an electronic circuit comprising: displaying a graphical user interface through which a user can initiate

generation of the API, Nilsson teaches on page 1, a program which provides and API for the use in simulating electronic circuits. With regard to claim 19, teaching selecting a component from a plurality of components for placement in the electronic circuit, the component for executing an implementable embedded function in the electronic circuit, Nilsson teaches, on pages 4 and 7, the process of selecting a component (place/part) and placing it in the schematic, the part having embedded functionality which is executed upon simulation/analysis. With regard to claim 19, further teaching configuring the selected component via the GUI, Nilsson teaches, on pages 4 and 5, the user configuring the selected part. With regard to claim 19, further teaching storing descriptive data relative to said selected component and said configuration, Nilsson teaches, on pages 4 and 5, the user configuring the selected part, adding descriptive data and clicking ok to store this information in association with the part. With regard to claim 19, teaching utilizing the interface to access the descriptive data, Nilsson teaches, on pages 4 and 5, double clicking on a part to see descriptive data. With regard to claim 19, further teaching initiating the GUI to invoke a processing of the descriptive data causing an automatic generation of the API, in response to a user input, the API for controlling the function of the component in the electronic circuit, Nilsson teaches, on pages 9-12, the system processing the descriptive data of the completed circuit and automatically generating an API (such as the net list, later displayed in Pspice A/D window (see page 7 and 11)), in response to the user input selecting the Pspice/Run menu option.

Nilsson teaches a system in which a graphical user interface is started with an application program, and upon a schematic circuit being drawn, via user input in the graphical user interface, a netlist is generated by analyzing the schematic circuit elements and their underling attributes, this netlist is then analyzed and provided for display on the GUI (see pages 7 and 11), however, he doesn't specifically teach an API comprising a device interface used to customize functionality of the electronic circuit; and interrupt activity framework for source programming and controlling the embedded function of the component in the electronic circuit, through user interaction with said new API. Lawman teaches a system for providing real time design of an electronic circuit via user input through a graphical user interface creating or modifying a logic circuit (see abstract and column 5, lines 30-42), similar to that of Nilsson, but further teaches that when a user adds a circuit element to the electronic circuit it causes corresponding elements to be added to the netlist, these take the form of API messages (forming program code based on functions assigned to components in the circuit diagram). After implementation, API messages are set back to the schematic entry system for display as feedback on the GUI (see column 5, lines 36-42 and column 6, lines 14-55). It would have been obvious to one of ordinary skill in the art, having the teachings of Nilsson and Lawman before him at the time the invention was made to modify electronic circuit design program of Nilsson to include the API functionality of Lawman. One would have been motivated to make such a combination because the systems implement a similar means of generating an interface via graphical

programming, Lawman further specifies user input causing an API to generate the GUI, via initiating processing, which would be an effective way of GUI generation for Nilsson.

Though it seems that the invention of Nilsson and Lawman modifies an API forming a new API (see column 6, lines 25-34 of Lawman), neither specifically state the forming of the new API. Williams teaches a system for designing a circuit through functional parts (see column 2, lines 56-65), similar to that of Nilsson and Lawman, but further teaches the creation of an API that defines the functional characteristics of the circuit and teaches allowing user manipulation of objects defined in the API (see column 2, line 56 through column 3, line 16). It would have been obvious to one of ordinary skill in the art, having the teachings of Nilsson, Lawman, and Williams before him at the time the invention was made to modify the circuit design systems of Nilsson and Lawman to include the actual creation of a new API, did Williams. One would have been motivated to make such a combination because this allows for greater portability of the designed circuit, in all references a user is creating a circuit design and allowing a user to interface with the created circuit, effectively creating an API (interface to an program representing the electronic circuit).

12. Claims 6, 15, and 24 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nilsson, Lawman, Williams, and Hsu, Patent #6,138,270.

13. With regard to claims 6, 15, and 24, which teach the API comprising assembly code, Nilsson and Lawman teach the system as described above where user simulate circuits through placing components in a schematic, but Nilsson and Lawman don't

suggest the use of the API comprising assembly code. Hsu teaches a system for simulating electronic circuits, similar to that of Nilsson, but further teaches, in column 2, lines 24-62, the use of assembly code for API files. It would have been obvious to one of ordinary skill in the art, having the teachings of Nilsson, Lawman, and Hsu before him at the time the invention was made to modify the circuit simulation system of Nilsson and Lawman to include the assembly code API files, as did McDonald. One would have been motivated to make such a combination because assembly is a conventional text based language.

14. Claims 8, 17, and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nilsson, Lawman, Williams, and McDonald et al., Patent #6,530,065, hereinafter McDonald.

15. With regard to claims 8, 17, and 26, which teach the data being substantially expressed in XML, Nilsson and Lawman teach the system as described above where user simulate circuits through placing components in a schematic, but Nilsson and Lawman don't suggest the use of the APIs comprising XML files. McDonald teaches a system for simulating electronic circuits, through placing them in a circuit configuration (see column 4, lines 39-57), similar to that of Nilsson and Lawman, but further teaches, in column 9, lines 62-67, the use of XML for API files. It would have been obvious to one of ordinary skill in the art, having the teachings of Nilsson, Lawman, and McDonald before him at the time the invention was made to modify the circuit simulation system of Nilsson to include the XML API files as did McDonald. One would have been motivated

to make such a combination because if the components are shared over the Internet HTML or XML are the standards.

Response to Arguments

16. The arguments filed on 1-24-2007 have been fully considered but they are not persuasive. Reasons set forth below.

17. The Applicants argue that the references do not teach or suggest in a method of application programming interface generation fro an electronic circuit, an API the includes a frame work for "source programming and controlling" an "embedded function" of a component of an electronic circuit through user interaction with the API.

18. In response the Examiner respectfully submits that the Nilsson, Lawman, and Williams references all teach a user is creating a circuit design and allowing a user to interface with the created circuit, effectively creating an API (interface to an program representing the electronic circuit). Programming is done graphically through the placement, arrangement, and settings of graphical circuit elements, to control the circuit through the assigned embedded functions of the electronic circuit components.

Nilsson teaches, on pages 4 and 7, the process of selecting a component (place/part) and placing it in the schematic, the part having embedded functionality which is executed upon simulation/analysis.

Lawman teaches a system for providing real time design of an electronic circuit via user input through a graphical user interface creating or modifying a logic circuit (see abstract and column 5, lines 30-42), such that that when a user adds a circuit element

to the electronic circuit it causes corresponding elements to be added to the netlist, these take the form of API messages (forming program code based on functions assigned to components in the circuit diagram). After implementation, API messages are set back to the schematic entry system for display as feedback on the GUI (see column 5, lines 36-42 and column 6, lines 14-55).

Williams teaches a system for designing a circuit through functional parts (see column 2, lines 56-65), similar to that of Nilsson and Lawman, but further teaches the creation of an API that defines the functional characteristics of the circuit and teaches allowing user manipulation of objects defined in the API (see column 2, line 56 through column 3, line 16).

19. The Applicants argue that Nilsson, Lawman, and Williams don't teach the contents on the independent claim or configuring the selected components comprises placing the selected component in the electronic circuit.

20. In response the Examiner respectfully submits that Nilsson teaches, on page 4, configuring the circuit by selecting a component (place/part) and placing it in the schematic. Further see the Examiners answer to arguments of claim 1.

21. The Applicants argue that Nilsson, Lawman, and Williams don't teach the contents on the independent claim or the configuring the selected component further comprising setting parameters of the selected component, the parameters relative to the function of the component and to the electronic circuit.

22. In response the Examiner respectfully submits that Nilsson teaches, on pages 4 and 5, the user configuring the selected part by setting parameters that are relative to a function of the component in the circuit. Further see the Examiners answer to arguments of claim 1.

23. The Applicants argue that Nilsson, Lawman, and Williams don't teach the contents on the independent claim or configuring the selected component comprising selecting pin values for connecting the component to the electronic circuit.

24. In response the Examiner respectfully submits that Nilsson teaches, on page 46, connecting the op amp to VCC+ and VCC-, which in turn correspond to power supplies displayed connected over ground. Further see the Examiners answer to arguments of claim 1.

25. The Applicants argue that Nilsson, Lawman, and Williams don't teach the contents on the independent claim or the API comprising header files.

26. In response the Examiner respectfully submits that Nilsson teaches, on page 12, the inclusion of a definition file. Further see the Examiners answer to arguments of claim 1.

27. The Applicants argue that Nilsson, Lawman, and Williams don't teach the contents on the independent claim or the API comprising include files.

28. In response the Examiner respectfully submits that Nilsson teaches, on page 12, the inclusion of an include file. Further see the Examiners answer to arguments of claim 1.

29. The Applicants argue that Nilsson, Lawman, and Williams don't teach the contents on the independent claim or the interface comprising a plurality of windows.

30. In response the Examiner respectfully submits that Nilsson teaches, on pages 5, 9, and 12, the interface comprising a plurality of windows. Further see the Examiners answer to arguments of claim 1.

31. The Applicants argue that Nilsson, Lawman, Williams, and Hsu don't teach the contents on the independent claim or "the new API further comprising assembly code files.

32. In response the Examiner respectfully submits that Hsu teaches a system for simulating electronic circuits, similar to that of Nilsson, but further teaches, in column 2, lines 24-62, the use of assembly code for API files. Further see the Examiners answer to arguments of claim 1.

33. The Applicants argue that Nilsson, Lawman, Williams, and McDonald don't teach the contents on the independent claim or wherein said data is substantially expressed in extensible markup language.

34. In response the Examiner respectfully submits that McDonald teaches a system for simulating electronic circuits, through placing them in a circuit configuration (see column 4, lines 39-57), similar to that of Nilsson and Lawman, but further teaches, in column 9, lines 62-67, the use of XML for API files. Further see the Examiners answer to arguments of claim 1.

Conclusion

35. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Dennis G. Bonshock whose telephone number is (571) 272-4047. The examiner can normally be reached on Monday - Friday, 6:30 a.m. - 4:00 p.m.

36. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kristine Kincaid can be reached on (571) 272-4063. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

37. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

3-22-07
dgb

RAYMOND J. BAYERL
PRIMARY EXAMINER
ART UNIT 2173